UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/727,055	12/04/2003	Kei Yoneda	56937-100	4830

7590 07/09/2008

MCDERMOTT, WILL & EMERY
600 13th Street, N.W.
Washington, DC 20005-3096

EXAMINER
BUI, HANH THI MINH

ART UNIT	PAPER NUMBER
2192	

MAIL DATE	DELIVERY MODE
07/09/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/727,055	YONEDA ET AL.	
	Examiner	Art Unit	
	HANH T. BUI	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *31 March 2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3,5,7,9,11,13,15,16,18-27 and 32-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3,5,7,9,11,13,15,16,18-27 and 32-36* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Status of Claims

1. Applicant's amendment dated March 31st, 2008 responding to the December 31st, 2007 Office Action provided in the rejection of claims 1-31.

Claims 1, 5, 11, 16, 18, 19, 21, 23 and 27 have been amended.

Claims 2, 4, 6, 8, 10, 12, 14, 17 and 28-31 have been canceled.

New Claims 32-36 have been added. Claims *1*, 3, 5, 7, 9, 11, 13, 15, 16, 18-27 and 32-36 are pending in the application, of which claims 1, 16, 18, 32, 35 and 36 are in independent form and which have been fully considered by the examiner.

The 35 USC 112 rejection of claims 1-31 has been withdrawn in view of Applicant's amendments to the claims.

2. Applicants' arguments filed on March 31st, 2008 based on amended and newly added claims, have been considered but are moot in view of the new ground(s) of rejection. See Hilla et al. (US Patent 7,155,722 – hereinafter, Hilla) in view of Husain et al. (Pub. No. US 2003/0126260 – hereinafter, Husain - art made of record) further in view of Trissel et al. (US Patent 5,274,815 - hereinafter, Trissel) and further in view of Potter (Patent No. 6,505,269 – hereinafter, Potter - art made of record).

REMARKS

3. Answers To Applicant's Arguments:

a. **Arguments:** Hilla does not disclose or suggest the recited "substituting step" or "equivalent process" recited therein (See Remarks Pg. 15: 6-7).

Answer: Husain discloses in Figure 4, 9 and the associated text; "… Swapping ***(substituting)*** the computer blade 401 ***(resource)*** with the computer blade 403 ***(resource)*** may involve a single computer ***switch*** from the first computer to the second computer … after the swap, the original user ***(equivalent process)*** of the computer blade 401 will be using the computer blade 403 and the original user ***(process)*** of the computer blade 403 will be using the computer blade 401" (emphasis added – See para. [0080]) and "… if the second computer ***(resource)*** is a higher performing computer, and the user ***(equivalent process)*** of the first computer needs more computational power than the user ***(process)*** of the second computer, the computers assigned to each user may be swapped ***(substitute)*** …" (emphasis added – See para. [0108-0111]).

It is noted that the user uses applications on the computer, therefore user can be interpreted as process used in applications.

It would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teaching of Husain into the teaching of Hilla because such combination would have provided improved systems and methods for managing resources in a system of networked computers as suggested by Husain (See para. [0009]).

b. **Arguments:** Hilla does not disclose that processor 302 can access the session block 420 "by using the same address" as recited in claim 16 (See Remarks Pg. 15: 17-18).

Answer: Potter discloses in FIG. 4 and the associated text, e.g., Col. 7: 27-40; "Ext Mem 400 comprising at least one synchronous dynamic random access memory (SDRAM) array organized into a ***plurality*** of (e.g., 4) ***banks (Banks 0-3)*** … is ***accessible*** by a column of processors ***(including first processor)*** … The memory arrays logically form a single address space ***(same address)***" – emphasis added.

c. **Arguments:** Hilla does not disclose or suggest "alter[ing] software processing processes," as recited in claim 16 (See Remarks Pg. 16: 7-8).

Answer: Examiner asserts that Hilla discloses in Col. 3: 29-50; "The access monitor is arranged to compile statistics from each processor resource of the pool and provides those statistics to a central load balancing resource for use when determining assignment of loads (tasks) to the various processor resources. The inventive mechanism is arranged to differentiate between active and inactive tasks ***(contention information)*** assigned to each processor resource. For example, a processor may have been assigned thousands of tasks, as in the case with mobile wireless traffic patterns, but the tasks or sessions may be idle ***(contention information)***. In this case, the processor may not be processing any information and, accordingly, is available to perform additional processing operations ***(altering)***. By ***monitoring*** certain memory cycles according to the inventive technique, the access monitor provides ***statistics*** to the central load balancer indicative of the ***actual***

measured activity of each processor resource. The central load balancer then only needs to keep track of the total number of sessions assigned to each processor resource to guard against reaching a predefined maximum load (task) limit. Then the load balancer may assign ***(alter)*** additional tasks based upon ***(in response)*** the measured activity ***(contention information)*** received ***(obtained by)*** from the access monitor." – emphasis added.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claim 27 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

6. Claim 27 recites the limitation "*the process-identifying process*" in line 2 and 5. There is insufficient antecedent basis for this limitation in the claim.

7. Claim 27 recites the limitation "*the compiler*" in line 3. There is insufficient antecedent basis for this limitation in the claim.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and

the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. Claims 1, 3, 5, 7, 9, 11, 13, 15 and 18-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hilla et al. (US Patent 7,155,722 – hereinafter, Hilla) in view of Husain et al. (Pub. No. US 2003/0126260 – hereinafter, Husain - art made of record).

Regarding claim 1:

Hilla discloses a load balancing mechanism and technique that monitors a memory interface associated with a processor resource in a processor pool.

- *a monitoring step, which monitors a status of use of a resource identified as used by a process for processor;*

(FIG. 3 and the associated text, e.g., Col. 5: 53-55; "the access monitor 500 is configured to **monitor the activity** (e.g., memory access requests) ***(status)*** over the memory interface ***(resource)***" - emphasis added).

- *a determining step, which determines the status of use of the resource based upon contention information obtained in the monitoring step;*

(Col. 3: 29-35; "The access monitor is arranged to compile statistics from each processor resource of the pool and provides those statistics to a central load balancing resource for use when ***determining assignment of loads*** (tasks) to the various processor resources. The inventive mechanism is arranged to differentiate between active and inactive tasks ***(contention information)*** assigned to each processor resource." – emphasis added.)

But, Hill does not explicitly teach:

- *a substituting step, which substitutes an equivalent process for a process using the resource, based upon the results of the determining step; wherein the equivalent process is for the processor, is equivalent to the process using the resource, and makes reduced use of the resource.*

However, Husain discloses in Figure 4, 9 and the associated text; "… Swapping ***(substituting)*** the computer blade 401 ***(resource)*** with the computer blade 403 ***(resource)*** may involve a single computer ***switch*** from the first computer to the second computer … after the swap, the original user ***(equivalent process)*** of the computer blade 401 will be using the computer blade 403 and the original user ***(process)*** of the computer blade 403 will be using the computer blade 401" (emphasis added – See para. [0080]) and "… if the second computer ***(resource)*** is a higher performing computer, and the user ***(equivalent process)*** of the first computer needs more computational power than the user ***(process)*** of the second computer, the computers assigned to each user may be swapped ***(substitute)*** …" (emphasis added – See para. [0108-0111]).

It is noted that the user uses applications on the computer, therefore user can be interpreted as process used in applications.

It would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teaching of Husain into the teaching of Hilla because such combination would have provided improved systems and methods for managing resources in a system of networked computers as suggested by Husain (See para. [0009])

Regarding claim 3:

Hilla and Husain disclose *the software processing method according to claim1, wherein:*

- *the resource is a storing device for a process, and the monitoring step for the status of use monitors the status of use of the storing device.*

(FIG. 3 and the associated text, e.g., Col 5: 28-29, emphasis added; "The memory 400 comprises ***storage locations addressable by the processor*** for storing software programs and data structures").

Regarding claim 5:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the monitoring step for status of use stores the preceding statuses of use of the storing device corresponding to a plurality of entries so that the contention information is generated based upon the stored status and current status of use.*

(FIG. 5 and the associated text; "The data structure is preferably an active session table 520 having a ***plurality of entries*** 522, each of which is associated with a corresponding session and configured to ***store status information, such as statistics, pertaining to activities associated with that session***" (emphasis added – See Col. 6: 52-56) and "the status information stored within each entry of the active session table represents ***activity (accesses)*** to the session associated with that entry within a

particular time interval. During a predefined interval, each access by a processor 302 to a particular session block 420 is ***recorded*** within the active session table 520 as an increment ***(preceding statuses)*** to the status information stored within the associated entry 522. The active session table 520 thus keeps track of which sessions had a 'hit' during the interval of time, thereby indicating an active session… the central load balancer interface logic 516 provides to the central load balancer 250 a count as to the number of active sessions its associated processor 302 processed during the ***previous time*** interval" (emphasis added – See Col. 7: 35-59)).

Regarding claim 7:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the monitoring step for status of use stores the time of use when the storing device is in use, and based upon whether or not the time of use is not less than a predetermined value, the contention information is generated.*

(FIG. 5 and the associated text, e.g., Col. 7: 16-34; "Over a period of time, certain active entries 522 may be ***'aged out'*** of the table 510 to allow insertion of more recent active sessions within those entries. The active session table 520 ***keeps track of the activity of sessions*** within the session region 410 of memory 400 during that time period. Upon being presented a decoded address over line 510 from the address decode logic 508, the session update logic 512 (i) updates the appropriate entry 522 (if it exists) by, e.g., incrementing status information contained within the entry or (ii)

creates a new entry of the table 520. Specifically, the session update logic 512 ***compares*** that address ***(predetermined value)*** with the address of a session stored within each entry 522 of the active session table 520. If there is not a match, the session update logic 512 allocates an entry by executing an aging algorithm to remove an existing entry from the table 520 and insert the current session identified by the decoded address into that entry. The aging algorithm may be based on, e.g., a count of the number of accesses made by a processor 302 to the session block/record 420 associated with the entry" - emphasis added).

Regarding claim 9:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the resource comprises a storing device for a process and a bus that connects the processor to the storing device, and the monitoring step for status of use monitors the status of use of the bus.*

(FIG. 3 and the associated text; "Each resource 300 preferably comprises a processor 302 coupled to a memory 400 via a ***memory bus or interface 310***. The memory 400 comprises ***storage locations*** addressable by the processor for storing software programs and data structures" (emphasis added – See Col. 5: 25-29) and "The memory interface 310 ***(bus)*** comprises a plurality of wires or "lines," including memory address, data and control lines … ***monitors*** the physical signals on the interface 310

(bus) to determine the active/inactive status of sessions distributed among the processor resources" (emphasis added – See Col. 5: 39-40)).

Regarding claim 11:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the monitoring step for status of use stores the preceding statuses of use of the bus corresponding to a plurality of entries so that the contention information is generated based upon the stored status and current status of use.*

(FIG. 5 and the associated text; "The access ***monitor*** comprises memory interface logic 502 coupled to the memory interface 310 and to address decode logic 508. The memory interface logic 502 provides a physical interface to the ***memory interface "bus" including buffers and transceivers that receive and output information over the bus***" (emphasis added – See Col. 6: 34-37)).

Regarding claim 13:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the monitoring step for status of use stores the time of use when the bus is in use, and based upon whether or not the time of use is not less than a predetermined value, the contention information is generated.*

(FIG. 5 and the associated text; "Over a period of time, certain active entries 522 may be '***aged out***' of the table 510 to allow insertion of more recent active sessions within those entries. The active session table 520 ***keeps track of the activity of sessions*** within the session region 410 of memory 400 during that time period. Upon being presented a decoded address over line 510 from the address decode logic 508, the session update logic 512 (i) updates the appropriate entry 522 (if it exists) by, e.g., incrementing status information contained within the entry or (ii) creates a new entry of the table 520. Specifically, the session update logic 512 ***compares*** that address ***(predetermined value)*** with the address of a session stored within each entry 522 of the active session table 520. If there is not a match, the session update logic 512 allocates an entry by executing an aging algorithm to remove an existing entry from the table 520 and insert the current session identified by the decoded address into that entry. The aging algorithm may be based on, e.g., a count of the number of accesses made by a processor 302 to the session block/record 420 associated with the entry" (emphasis added – See Col. 7: 16-34)).

Regarding claim 15:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the resource is a second processor that executes a process in response to a processing request from the processor, and the monitoring step for status of use monitors the status of use of the second processor.*

(FIG. 2 and the associated text, e.g., Col. 4: 58-60; "The intermediate network node 200 comprises a **multiprocessor** environment of processor resources (PS) 300 organized as a 'processor pool 210'" - emphasis added.

Examiner notes that the processor pool 210 comprises a plurality of processor resource 300, therefore the resource can be another/second processor that executes a process in response to a processing request from the processor.

FIG. 3 and the associated text, e.g., Col 5: 24-29; "FIG. 3 is a schematic block diagram of a processor resource 300 ***(second processor)*** within the processor pool 210. Each resource 300 ***(second processor)*** preferably comprises a processor 302 coupled to a memory 400 via a memory bus or interface 310. The memory 400 comprises storage locations addressable by the processor for storing software programs and data structures" - emphasis added

Col. 3: 29-33; "The access ***monitor*** is arranged to compile statistics ***(status)*** from each processor resource of the pool" - emphasis added).

Regarding claim 18:

Hilla discloses a load balancing mechanism and technique that monitors a memory interface associated with a processor resource in a processor pool.

- *a monitoring step, which monitors a status of use of a resource identified as used by a process for processor;*

(FIG. 3 and the associated text, e.g., Col. 5: 53-55; "the access monitor 500 is configured to **monitor the activity** (e.g., memory access requests) ***(status)*** over the memory interface ***(resource)***" - emphasis added).

- *a determining step, which determines the status of use of the resource based upon contention information obtained in the monitoring step;*

(Col. 3: 29-35; "The access monitor is arranged to compile statistics from each processor resource of the pool and provides those statistics to a central load balancing resource for use when ***determining assignment of loads*** (tasks) to the various processor resources. The inventive mechanism is arranged to differentiate between active and inactive tasks ***(contention information)*** assigned to each processor resource." – emphasis added.).

- *a storing for storing contention information obtained in the monitoring step at a current time;*

(FIG. 3 and the associated text, e.g., Col. 5: 53-60; "the access monitor 500 is configured to **monitor the activity** (e.g., memory access requests) ***(status)*** over the memory interface ***(resource)***. These certain regions are configured to maintain status information associated with sessions stored in the memory. By monitoring the activity associated with particular session, the access monitor 500 may gather status information, such as statistics." - emphasis added).

But, Hill does not explicitly teach:

- *a substituting step, which substitutes an equivalent process for a process using the resource, based upon the results of the determining step; wherein the*

equivalent process is for the processor, is equivalent to the process using the resource, and makes reduced use of the resource.

However, Husain discloses in Figure 4, 9 and the associated text; "… Swapping ***(substituting)*** the computer blade 401 ***(resource)*** with the computer blade 403 ***(resource)*** may involve a single computer ***switch*** from the first computer to the second computer … after the swap, the original user ***(equivalent process)*** of the computer blade 401 will be using the computer blade 403 and the original user ***(process)*** of the computer blade 403 will be using the computer blade 401" (emphasis added – See para. [0080]) and "… if the second computer ***(resource)*** is a higher performing computer, and the user ***(equivalent process)*** of the first computer needs more computational power than the user ***(process)*** of the second computer, the computers assigned to each user may be swapped ***(substitute)*** …" (emphasis added – See para. [0108-0111]).

It is noted that the user uses applications on the computer, therefore user can be interpreted as process used in applications.

It would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teaching of Husain into the teaching of Hilla because such combination would have provided improved systems and methods for managing resources in a system of networked computers as suggested by Husain (See para. [0009])

Regarding claim 19:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the contention information is processing time from the issuance of the processing request for the resource until the completion of the process, and the determining process for the status of use is a process which compares the processing time to a preset value.*

(Col. 7: 60 - Col 8: 15; “the central load balancer 250 analyzes the status information provided by each access monitor 500 of a processor resource 300 to determine the actual load ***(processing time)*** executed by the resource during a specified period of time…It will be understood to those skilled in the art that various forms of status information may advantageously be gathered in accordance with the principles of the present invention. The types of statistics ***(determining process)*** that may be collected include the number of active sessions during a predetermined time interval ***(preset value)***, the number of "hits" (accesses) to a particular session block during a time interval and the number of creations/destroys of sessions during that interval.” – emphasis added.).

Regarding claim 20:

The rejection of base claim 18 is incorporated. All the limitations of this claim have been noted in the rejection of claim 19.

Regarding claim 21:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the contention information is waiting time from the issuance of the processing request for the resource until the start of the process, and the determining process for the status of use is a process which compares the waiting time to a preset value.*

(Col 8: 3-15; "It will be understood to those skilled in the art that various forms of status information ***(contention information)*** may advantageously be gathered in accordance with the principles of the present invention. The types of statistics ***(determining process)*** that may be collected include the number of active sessions during a predetermined time interval ***(preset value)***, the number of "hits" (accesses) to a particular session block during a time interval and the number of creations/destroys of sessions during that interval ***(waiting time)***" - emphasis added).

Regarding claim 22:

The rejection of base claim 18 is incorporated. All the limitations of this claim have been noted in the rejection of claim 21.

Regarding claim 23:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the determining process for status of use reexamines the determination for the status of use of the resource regularly or irregularly.*

(Col. 7: 64-67; "the central load balancer may overlay the number of active sessions per processor onto the total number of sessions assigned per processor to determine the type of activity ***(regular)*** versus inactivity ***(irregularly)*** (quiescence) across the pool of resources" - emphasis added)

Regarding claim 24:

The rejection of base claim 18 is incorporated. All the limitations of this claim have been noted in the rejection of claim 23.

Regarding claim 25:

Hilla and Husain disclose *the software processing method according to claim 1, wherein:*

- *the determining process for status of use reexamines the determination for the status of use of the resource by using random numbers.*

(FIG. 5 and the associated text, e.g., Col. 6: 37-45; "The logic 502 also provides address lines 504 and control lines 506 ***(random numbers)*** to the address decode logic 508. The address decode logic 508 is configured with conventional circuitry adapted to decode the information presented over the lines 504, 506 in order to ***determine*** the operations performed by the processor 302 for a specific session" - emphasis added).

Regarding claim 26:

The rejection of base claim 18 is incorporated. All the limitations of this claim have been noted in the rejection of claim 25.

10. **Claim 16 and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hilla in view of Potter (Patent No. 6,505,269 – hereinafter, Potter – art made of record)**

Regarding claim 16:

Hilla discloses *a software processing method comprising:*

- *a monitoring step for status of use, which monitors the status of use of a second processor, the second processor performing processing in response to a processing request by a first processor;*

(FIG. 2 and the associated text, e.g., Col. 4: 58-60; "The intermediate network node 200 comprises a **multiprocessor** environment of processor resources (PS) 300 organized as a 'processor pool 210'" - emphasis added.

Examiner notes that the processor pool 210 comprises a plurality of processor resource 300, therefore the resource can be another/second processor that executes a process in response to a processing request from the processor.

FIG. 3 and the associated text, e.g., Col 5: 24-29; "FIG. 3 is a schematic block diagram of a processor resource 300 ***(second processor)*** within the processor pool 210. Each resource 300 ***(second processor)*** preferably comprises a processor 302 coupled to a memory 400 via a memory bus or interface 310. The memory 400

comprises storage locations addressable by the processor for storing software programs and data structures" - emphasis added

Col. 3: 29-33; "The access ***monitor*** is arranged to compile statistics ***(status)*** from each processor resource of the pool" - emphasis added).

- *an altering step for software processes, which alters software processing processes executed by the first processor or the second processor in response to contention information, the contention information being obtained in the monitoring step for status of use*

(Col. 3: 29-50; "The access monitor is arranged to compile statistics from each processor resource of the pool and provides those statistics to a central load balancing resource for use when determining assignment of loads (tasks) to the various processor resources. The inventive mechanism is arranged to differentiate between active and inactive tasks ***(contention information)*** assigned to each processor resource. For example, a processor may have been assigned thousands of tasks, as in the case with mobile wireless traffic patterns, but the tasks or sessions may be idle ***(contention information)***. In this case, the processor may not be processing any information and, accordingly, is available to perform additional processing operations ***(altering)***. By ***monitoring*** certain memory cycles according to the inventive technique, the access monitor provides ***statistics*** to the central load balancer indicative of the ***actual measured activity of each processor resource***. The central load balancer then only needs to keep track of the total number of sessions assigned to each processor resource to guard against reaching a predefined maximum load (task) limit. Then the

load balancer may assign ***(alter)*** additional tasks based upon ***(in response)*** the measured activity ***(contention information)*** received ***(obtained by)*** from the access monitor." – emphasis added).

But, Hilla does not explicitly teach:

- *wherein the first processor can access a plurality of memory banks by using a same address, and the plurality of memory banks includes a memory bank used for the first processor;*

However, Potter discloses in FIG. 4 and the associated text, e.g., Col. 7: 27-40; "Ext Mem 400 comprising at least one synchronous dynamic random access memory (SDRAM) array organized into a ***plurality*** of (e.g., 4) ***banks (Banks 0-3)*** … is ***accessible*** by a column of processors ***(including first processor)*** … The memory arrays logically form a single address space ***(same address)***" – emphasis added.

- *the contention information that is a signal that indicates memory bank switching from one of the plurality of memory banks to the memory bank used for the first processor.*

However, Potter further discloses "A first processor … can access a random location … at absolute time N ***(indication of a signal)***… the second processor may access the same ... location ***(same address)*** … at time N+7 ***(indication of a signal to switch)*** without contending with the first processor …" (emphasis added – See Col. 2: 45-54), "approximately 70 nsecs ***(indication of a signal)*** are required to completely cycle through a random access operation to a memory bank of the SDRAM resource" (emphasis added – See Col. 7:46-58) and "the minimum time required for a random

access to a location in the memory bank is approximately seven (7) cycles ***(signal)*** before another ***(switch)*** random access operation can be issue to that bank" (emphasis added – See Col. 8: 9-11).

It would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teachings of Potter into the teachings of Hilla because such combination would have provided a technique that enables fast and efficient accesses by processors of a symmetric multiprocessor system to contiguous storage locations of a memory resource as suggested by Potter (See Col. 2: 20-22)

Regarding claim 35:

This is another system version of the rejected claim 16 above, wherein all the limitations of this claim have been noted in the rejection of claim 16.

11. Claim 27 is rejected under 35 U.S.C. 103(a) as being unpatentable over Hilla in view of Husain and further in view of Trissel et al. (US Patent 5,274,815 - hereinafter, Trissel).

Regarding claim 27:

Hilla and Husain disclose *the software processing method according to claim 18 wherein:*

- *in the case when processes to be extracted by the process-identifying process are extracted from a plurality of portions of the software, the compiler further adds to the software an identifying process for identifying the portions of appearance of*

the processes identified by the process-identifying process, and the storing process stores the contention information for each of the portions of appearance so that the determining process for status of use carries out the determination by using the contention information stored for each of the portions of appearance.

(Col. 8: lines 26-36, emphasis added; "each field 422 of the record 420 is software **configurable** (given the static nature of the record format) such that a particular field may be **programmed** to reflect activity used to determine balancing of sessions across the pool of processors. The hardware assist device 500 **cooperates with the software executing on the processor** 302...")

But, Hilla and Husain do not explicitly teach

- *the compiler*

Trissel further discloses a dynamic instruction modifying controller and operation method, wherein the background teaches more clearly about the use of compiler "The most primitive way to alter a computer program is to edit the computer program, make all modifications desired by changing instructions in the computer program, ***compile***, link, and execute the code to observe changes … ***Compiler*** options are statements or commands which alter the course of the compilation process such as to determine optimization levels and enable debugging modes or facilities, and/or gather statistical information. By changing, deleting, or adding compiler options, a user or a programmer can alter the performance, execution flow, or results of a computer program. ***Compiler*** options are widely used because they are, in most cases, easier to implement than other technologies" (emphasis added - See Col. 1: 25-40).

It would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teachings of Trissel into the teachings of Hilla and Husain because such combination would have achieved flexibility and control of software as suggested by Trissel (See Col. 1: lines15-16).

12. Claims 32-34 and 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hilla in view of Potter and further in view of Trissel et al. (US Patent 5,274,815 - hereinafter, Trissel).

Regarding claim 32:

Hilla and Potter disclose *the software processing method according to claim 16, further comprising the steps of:*

- *identifying a process which uses the second processor in the software;*

(Hilla further discloses in FIG. 2 and the associated text, e.g., Col. 4: lines 58-60; "The intermediate network node 200 comprises a **multiprocessor** environment of processor resources (PS) 300 organized as a 'processor pool 210'" - emphasis added.

Examiner notes that the processor pool 210 comprises a plurality of processor resource 300, therefore the resource can be another/second processor that executes a process in response to a processing request from the processor.)

- *mapping a process of using the first processor to the memory bank used for the first processor, and a process of using the second processor to a memory bank used for the second processor*

(Potter further discloses in FIG. 8 and the associated text, e.g., Col.11: 2-14; "FIG. 8 is a schematic block diagram of the XRAM controller 800 comprising address

mapping logic (i.e., ***address mapper*** 810) … The ***address mapper*** receives relevant address bits and mode select bits from the TMC processors … and converts them into interface (I/F) select, bank select, row and column bits … ***determine the particular bank and array to which the memory access operation is directed***" – emphasis added.

- *wherein the signal that indicates memory bank switching indicates switching to the memory bank used for the first processor.*

Potter further discloses "A first processor … can access a random location … at absolute time N ***(indication of a signal)***… the second processor may access the same ... location ***(same address)*** … at time N+7 ***(indication of a signal to switch)*** without contending with the first processor …" (emphasis added – See Col. 2: 45-54), "approximately 70 nsecs ***(indication of a signal)*** are required to completely cycle through a random access operation to a memory bank of the SDRAM resource" (emphasis added – See Col. 7:46-58) and "the minimum time required for a random access to a location in the memory bank is approximately seven (7) cycles ***(signal)*** before another ***(switch)*** random access operation can be issue to that bank" (emphasis added – See Col. 8: 9-11).

But Hilla and Potter do not explicitly teach:

- *compiling a software*

However, Trissel discloses a dynamic instruction modifying controller and operation method, wherein the background teaches more clearly about the use of compiler "The most primitive way to alter a computer program is to edit the computer

program, make all modifications desired by changing instructions in the computer program, ***compile***, link, and execute the code to observe changes… ***Compiler*** options are statements or commands which alter the course of the compilation process such as to determine optimization levels and enable debugging modes or facilities, and/or gather statistical information. By changing, deleting, or adding compiler options, a user or a programmer can alter the performance, execution flow, or results of a computer program. ***Compiler*** options are widely used because they are, in most cases, easier to implement than other technologies" (emphasis added - See Col. 1: lines 25-40).

It would have been obvious to one having ordinary skill in the art at the time of the invention was made to combine the teachings of Trissel into the teachings of Hilla and Potter because such combination would have achieved flexibility and control of software as suggested by Trissel (See Col. 1: lines15-16).

Regarding claim 33:

Hilla, Potter and Trissel disclose *the software processing method according to claim 32, further comprising the step of:*

- *locking an operation of memory bank switching so that a memory bank switching cannot take place in case the first processor or the second processor is performing processing.*

Potter further discloses "after the read (or write) command a parameter may be specified that 'closes' the bank ***(locking)*** …" (emphasis added – See Col. 7: 64- Col. 8: 8)

Regarding claim 34:

Hilla, Potter and Trissel disclose *the software processing method according to claim 33, further comprising the step of:*

- *unlocking the locked operation when the first processor or the second processor has finished the processing so that the signal that indicates memory bank switching can be accepted.*

Potter further discloses "In the case of random read access operation to the bank, an active command is issued … The active command functions a s a conventional read access strobe (RAS) command that 'opens' the bank ***(unlocking)***" (emphasis added – See Col. 7: 55-57)

Regarding claim 36:

The rejection of base claim 35 is incorporated. All the limitations of this claim have been noted in the rejection of claim 32.

Conclusion

13. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

14. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Hanh T. Bui whose telephone number is (571) 270-1976. The examiner can normally be reached on 9:30 AM - 4:30PM / Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/H. T. B./

Examiner, Art Unit 2192

/Tuan Q. Dam/

Supervisory Patent Examiner, Art Unit 2192